

# On the genericity properties in networked estimation: Topology design and sensor placement

Mohammadreza Doostmohammadian and Usman A. Khan

## Abstract

In this paper, we consider networked estimation of linear, discrete-time dynamical systems monitored by a network of agents. In order to minimize the power requirement at the (possibly, battery-operated) agents, we require that the agents can exchange information with their neighbors only *once per dynamical system time-step*; in contrast to consensus-based estimation where the agents exchange information until they reach a consensus. It can be verified that with this restriction on information exchange, measurement fusion alone results in an unbounded estimation error at every such agent that does not have an observable set of measurements in its neighborhood. To overcome this challenge, state-estimate fusion has been proposed to recover the system observability. However, we show that adding state-estimate fusion may not recover observability when the system matrix is structured-rank ( $S$ -rank) deficient.

In this context, we characterize the state-estimate fusion and measurement fusion under both full  $S$ -rank and  $S$ -rank deficient system matrices. The main results of this paper are the following. *Firstly*, we show that when the system matrix has full  $S$ -rank, state-estimate fusion alone (with no measurement fusion) can recover the observability. Subsequently, we characterize the minimal topology for inter-agent communication required for a stable networked estimator. *Secondly*, we provide methodologies to recover (networked) estimator observability when the system matrix is  $S$ -rank deficient. In particular, we introduce a novel agent classification based on their local measurements and identify the agents that are crucial for stable estimation error. We then provide topology modifications and sensor placement techniques to recover observability in the  $S$ -rank deficient scenario. Finally, we provide an iterative method to compute the local estimator gain at each agent that results into a stable estimation error once the observability is ensured using the aforementioned techniques.

**Keywords:** Networked estimation, Observability, Structured system theory, Generic rank

## I. INTRODUCTION

Estimation of dynamical systems with observations distributed among a network of agents is an important field of research, where the idea is to assign a group of agents to monitor a certain system

Department of Electrical and Computer Engineering, Tufts University, {mrd, khan}@ece.tufts.edu.

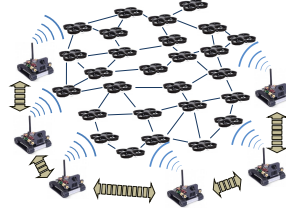


Fig. 1. A group of mobile ground-robots are tracking a large flock of quad-copters. Every ground-robot is monitoring part of the flock, and then shares its partial data with other ground-robots through a communication network.

or phenomenon of interest. Agents are distributed in the sense that each agent can only measure some of the states of a dynamical system, referred to as local measurements. For example, a group of sensors spread geographically over a large region to monitor daily temperature evolution. The measurement data and dynamical models are further corrupted by noise and disturbances. The objective is to enable each agent to make an unbiased decision on the global state relying only on its own measurement and the measurements from its immediate neighbors. Such a scheme is often referred to as *networked estimation* where the term network implies that the information is restricted on a sparse network.

Networked estimation is preferable to a wide range of applications as it is scalable and further requires less communication load at each individual agent, in contrast to the centralized case where each agent may require repeated long-distance communication to a central location. Applications of networked estimation include social networks [1] to learn global beliefs based on partial understanding of the state of the society, market, politics, etc., monitoring physical processes and environmental spatio-temporal fields [2], [3], state estimation in power systems [4]–[6], and multi-agent systems such as collaborative target tracking and flocking of mobile robots [7], see for example Fig. 1.

A variety of solutions exists for networked estimation starting from the earlier work in [8], [9] and references therein on parallel Kalman filter architectures for all-to-all connected networks, to more recent diffusion-based schemes via least mean square implementation, such as in Kalman filtering and smoothing [10] and distributed binary detection [11]. Meanwhile, incremental adaptive distributed strategies can be found in [12], [13] along with distributed moving horizon estimation [14] to minimize estimation error variance for constrained problems. State estimators based on low-cost single-bit data transmission is proposed in [15] with binary sign of innovations (sign of difference of measurement and estimated value). Information theoretic approach based on consensus over the Kullback-Leibler average of Gaussian PDFs is exploited in [16]. The literature can also be classified into static and dynamic estimation. In static

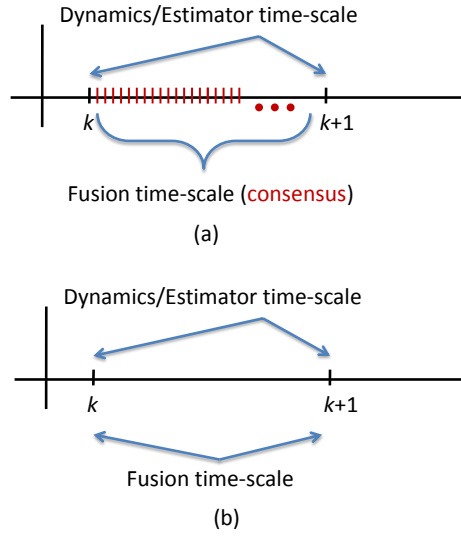


Fig. 2. (a) The traditional two-time scale consensus-based approach. (b) single time-scale approach.

estimation [2], [10]–[12], [17], [18], the target state to be estimated does not change over time, while dynamic estimation [4], [8], [14]–[16], [19]–[23] takes into account the time-evolution of the system<sup>1</sup>.

Consensus-based strategies have recently found a lot of interest in the context of sparsely-connected networks, where the main focus is to reduce the uncertainty of individual estimates by averaging on collaborative data. Early work in [19]–[23] considers a two time-scale method where consensus is implemented at a time-scale different than the system dynamics. These results require that a consensus is reached within every two time-steps of the system dynamics, and is thus, challenged with a large number (infinite, in general) of consensus iterations between every two steps of the dynamics. To elaborate this, consider Fig. 2(a), where a large number ( $\rightarrow \infty$ ) of data fusion iterations are implemented between every two successive time-steps,  $k$  and  $k+1$ , of the dynamics. This approach requires communication over a much faster rate than the sampling of the dynamics, and thus, in general, becomes practically infeasible when the underlying system is operating under power constraints and has restricted communication and computation budgets.

In contrast to the two time-scale approach to distributed estimation, recently References [1], [16]–[18], [23]–[25] studied the behavior of networked estimators when the communication time-scale is the same

<sup>1</sup>As stated in [17], diffusion algorithms can be extended for non-stationary (dynamic) tracking when the target is not moving too fast, i.e. its state is relatively stationary over a period such that the algorithm can converge.

as the time-scale of the dynamics, as shown in Fig. 2(b). This method is practically feasible for real-time applications and computationally efficient as compared to the two time-scale approach. A preliminary study on this single time-scale estimator is carried out in [24], where it is shown that a particular linear networked estimator has a bounded estimation error if the two-norm of the system matrix is less than the network tracking capacity—a function of the communication network and observation models. Notice that in the two time-scale method, the communication network becomes irrelevant due to more information exchanges among the individuals (the information in a sparsely connected graph is equivalent to the information in a fully connected graph when a large number of information exchanges are carried out). Therefore, the performance and properties of the underlying estimator depends only on the data fusion principles among the agents. However, in the single time-scale scenario of Fig. 2 (b), the underlying agent network remains sparse and an arbitrary communication network may not suffice to make the networked estimation error stable (e.g., see [24], [26], [27]).

In this context, the key problem is to design the structure of the inter-agent communication according to the underlying fusion rules in order to recover the observability of the networked estimator. In this paper, we use a variant of the Networked Kalman-type Estimator (NKE) protocol, initially introduced in [24]. The main contribution is to determine the communication network among the agents to recover the observability of the underlying estimation protocol, given that each agent may not be locally (in its neighborhood) observable. We study the observability with a structural point of view [28]–[34] in the sense that we explore the generic properties of the system. The generic properties are applicable to any choice of system parameters as long as the sparsity structure (zeros and non-zeros) is not violated. The generic approach is helpful when the underlying system parameters may change depending on the system operating point (linearization of non-linear dynamics) and is further significant in communication network design as the approach is independent of the exact value of the weights chosen for data fusion. Moreover, this implies that for smooth non-linear systems with fixed structure Jacobian matrix, similar analysis of the networked observability can be applied.

Comparing with other work in the literature, we consider single time-scale estimation, as opposed to the multi time-scale estimation in [19]–[22] and the *vanishing time-step* algorithms proposed in [35], [36]. Unlike [37], [38], we do not impose an agent hierarchy (i.e., we assume the processing/communication duties at all agents are the same). Avoiding agent hierarchy increases the reliability of node/link failure. We further do not require the communication network to be (strongly) connected [10], [11], [16], [17], [20], [26] or for it to include a cyclic path [12], [18]. Our goal is to design the network with minimal communication. Specifically, we use methodologies that are independent of exact system values and rely

only on the *structure* of the underlying system. This leads to a robust estimator design where the analysis is not algebraic, as in the conventional Grammian or PBH observability tests, but graph-theoretic [30], [31].

We now describe the rest of the paper. Section II provides preliminary material on basic dynamical system estimation and structured systems theory, whereas Section III presents our problem formulation. Section IV enlists our assumptions and describes a novel agent classification method. Section V covers the main results of this paper on state and output fusion, whereas local gain design is explored in Section VI. We provide an illustrative example and simulations in Section VII, and finally, Section VIII concludes the paper.

## II. BACKGROUND AND PRELIMINARIES

We consider the system model to be a discrete-time linear dynamical system:

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + \mathbf{v}_k, \quad (1)$$

where  $\mathbf{x}_k \in \mathbb{R}^n$  is the state vector,  $A = \{a_{ij}\} \in \mathbb{R}^{n \times n}$  is the system matrix, and  $\mathbf{v}_k \sim \mathcal{N}(0, V)$  is the system noise. We note that the main emphasis of this paper is not on modeling but on structural characteristics of the underlying system and the results we present hold for any phenomenon following Eq. (1). We assume that the dynamical system is monitored by a network of  $N$  agents such that each sensor  $i$  has the following observation model:

$$\mathbf{y}_k^i = C_i \mathbf{x}_k + \mathbf{r}_k^i, \quad (2)$$

where  $\mathbf{y}_k^i \in \mathbb{R}^{p_i}$  is the output vector at agent  $i$ ,  $\mathbf{r}_k^i \sim \mathcal{N}(0, R_i)$  is the output noise, and  $C_i$  is the output matrix at agent  $i$ . With this notation, we can write the global observation model as

$$\mathbf{y}_k = C\mathbf{x}_k + \mathbf{r}_k, \quad (3)$$

where

$$\mathbf{y}_k = \begin{bmatrix} \mathbf{y}_k^1 \\ \vdots \\ \mathbf{y}_k^N \end{bmatrix}, \quad C = \begin{bmatrix} C_1 \\ \vdots \\ C_N \end{bmatrix}, \quad \mathbf{r}_k = \begin{bmatrix} \mathbf{r}_k^1 \\ \vdots \\ \mathbf{r}_k^N \end{bmatrix}, \quad (4)$$

$\mathbf{r}_k \sim \mathcal{N}(0, R)$  is the global observation noise with  $R = \text{blockdiag}[R_1, \dots, R_N]$ , and  $C = \{c_{ij}\}$  is the global output matrix.

### A. Centralized estimator

Let  $\hat{\mathbf{x}}_{k|k}^c$  be the centralized Kalman estimator [39] at time  $k$  given all the observations,  $\mathbf{y}_k$ , up to time  $k$ . It can be shown that the error,  $\hat{\mathbf{e}}_{k|k}^c = \mathbf{x}_k - \hat{\mathbf{x}}_{k|k}^c$ , in this estimator is given by

$$\hat{\mathbf{e}}_{k|k}^c = (A - K_c C A) \hat{\mathbf{e}}_{k-1|k-1}^c + \eta_k, \quad (5)$$

where  $K_c$  is the centralized Kalman gain and the vector  $\eta_k$  collects the remaining terms that are independent of  $\hat{\mathbf{e}}_{k-1|k-1}^c$ . It is well known that the centralized Kalman error,  $\hat{\mathbf{e}}_{k|k}^c$  is stable if and only if all the unstable modes (eigenvalues) of the system are observable. For the ease of explanation, we assume that there are no stable unobservable nodes. In other words, detectability and observability are equivalent throughout this paper.

In the traditional sense of  $n$ -step  $(A, C)$ -observability, the observability Gramian is given by

$$\mathcal{O} = [C^T \ A^T C^T \ \dots \ (A^{n-1})^T C^T]. \quad (6)$$

Algebraic tests for observability check the Gramian,  $\mathcal{O}$ , for being full-rank or the matrix  $\mathcal{O}^T \mathcal{O}$  for being invertible. An alternative method is the PBH (Popov-Belevitch-Hautus) observability test [40], which requires the matrix,  $[A^T - sI \ C^T]$ , to be full-rank for all  $s$ . The matrix  $[A^T - sI]$  is full rank for all values of  $s$  other than the eigenvalues of  $A$  and, therefore, the PBH test is needed to be checked *only* for these values.

Note that, both these algebraic methods rely on the knowledge of exact values of each element in the matrices  $A$  and  $C$ . However, in many dynamical systems, only the sparsity (zero and non-zero pattern) of these matrices may remain fixed while the non-zero elements are subject to change. For example, when the elements of the concerned matrices depend on certain parameters or operating points. Hence, these conventional methodologies fail to check for observability in such cases and graph-theoretic techniques are to be employed. We introduce such graph-based methods below.

### B. Graph notations

Let  $X = \{x_1, \dots, x_n\}$  denote the state set, and let  $Y = \{y_1, \dots, y_p\}$  denote the output set. We define the *system digraph* as  $G_A = (V, E)$ , where  $V = X \cup Y$  is the vertex set, and  $E$  is the edge set containing directed edges,  $(v_1, v_2) \in E$ , of the form  $v_1 \rightarrow v_2$  with  $v_1, v_2 \in V$ . The edge set  $E$  is defined as  $E_A \cup E_C$ , where  $E_A = \{(x_j, x_i) \mid a_{ij} \neq 0\}$  and  $E_C = \{(x_j, y_i) \mid c_{ij} \neq 0\}$ . A *path* of length  $\ell$  from  $v_1 \in V$  to  $v_\ell \in V$  is such that there exists a sequence of vertices,  $v_1, v_2, \dots, v_\ell$  with each subsequent edge,  $(v_1, v_2), (v_2, v_3), \dots, (v_{\ell-1}, v_\ell) \in E$ . Here  $v_1$  is the *begin-vertex* of the path and  $v_\ell$  is

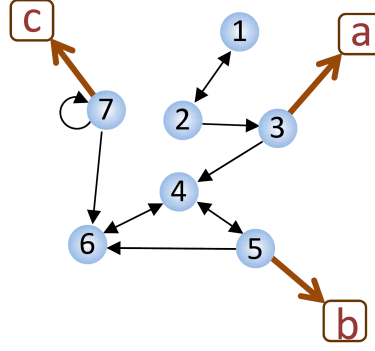


Fig. 3. This digraph is an example of  $(A, C)$  observable system based on the Theorem 1.

its *end-vertex*. Here, we assume that each vertex contained in a path occurs only once (*simple path*). A path is said to be *Y-topped* if it ends at a vertex in  $Y$ . A digraph is called strongly connected if there exist a directed path from each vertex to every other vertex in the digraph. In a not strongly connected digraph, define *Strongly Connected Components (SCC)* as its maximal strongly connected partitions or sub-graphs. A *cycle* is a simple path where the begin and end vertices are the same. Since the nodes in  $Y$  have no outgoing link, nodes included in a cycle all belong to  $X$ . As an example, consider Fig. 3 which shows the system digraph of a dynamical system with  $n = 7$  states (encircled) and  $N = 3$  measurements (or agents) denoted by squares. We now introduce some new concepts on SCCs over state vertices. These will be helpful in describing our results.

*Definition 1 (Parent SCC):* A state SCC, is called a *parent SCC*, if it has no outgoing link to any state vertex not belonging to itself.

*Definition 2 (Child SCC):* Any SCC that is not a parent SCC is a *child SCC*.

Notice that the set of disjoint state SCCs in system matrix  $A$  can be explicitly characterized as either a parent or a child. As an example, the SCC containing vertices  $\{4, 5, 6\}$  in Fig. 3 is a parent SCC, since there is no outgoing edges from its states to other states  $\{1, 2, 3, 7\}$  not included in it. Furthermore,  $\{1, 2\}$  and  $\{7\}$  are child SCCs. More details on parent/child SCCs and efficient algorithms for computing SCCs in a digraph can be found in [27] and [41], respectively. We now use the concepts from this section to formally introduce structured systems theory and generic properties.

### C. Structured systems theory

Structural analysis deals with system properties that do not depend on the numerical values of the parameters but only on the underlying structure (zeros and non-zeros) of the system [28]–[34]. It turns out that if a structural property is true for one *admissible* choice of non-zero elements as free parameters it is true for *almost all* choices of non-zero elements and, therefore, is called a *generic* property of the system [42]. Furthermore, it can be shown that those particular (non-admissible) choices for which the generic property does not hold lie on some algebraic variety with zero Lebesgue measure, for more details see [42], [43].

*Definition 3 (S-rank):* The structural rank (also called generic rank) is the maximum rank for all numerical values of the non-zero entries of the matrix  $A$ . It is, in fact, an upper-bound on the numerical rank of  $A$ .

The  $S$ -rank as a generic property holds for almost all choices of nonzero parameters of the matrix,  $A$ . It is equal to the cardinality of the maximum matching associated to the *bipartite graph* associated to the matrix,  $A$ . In the algebraic sense, this is the maximum number of non-zero elements in distinct rows and columns of the matrix,  $A$  [44]. Details on the generic rank implication in graph theoretic sense and algorithms on maximum matching can be found in [30], [41]. Among other generic properties, controllability/observability are of interest in the context of this paper, see [29], [30], [33], [34], [45], [46] for details. We extend the following theorem from the generic controllability results in [31].

*Theorem 1:* A dynamical system is generically observable if and only if in the system digraph:

- (i) Every state is the begin-node of a path that ends in an output (termed as a  $Y$ -topped path);
- (ii) There exist a disjoint union of  $Y$ -topped paths and cycles that cover all the state vertices.

The following lemma is from [28].

*Lemma 1:* The condition (ii) in Theorem 1 on the generic observability of  $(A_{n \times n}, C_{m \times n})$  is equivalent to the following:

$$S\text{-rank} \left( \begin{bmatrix} A \\ C \end{bmatrix} \right) = n. \quad (7)$$

The proof of Theorem 1 for generic controllability and Lemma 1 is given in [28], [31], where other equivalent graph-theoretic conditions to generic controllability (observability) are also defined that we omit here. As an example, consider the system shown in Fig. 3. It can be verified that each state is a begin-vertex of a  $Y$ -topped path, and  $\{7\}$ ,  $\{4, 5, 6\}$ ,  $\{1, 2, 3, a\}$  constitute a disjoint union of cycles and  $Y$ -topped paths that cover all the state vertices in  $X$ . Thus, satisfying both conditions in Theorem 1 and the system in Fig. 3 is observable for almost all choices of non-zero elements.



#### D. Corresponding graphs

In this paper, we deal with two different graph representations: *system digraph*,  $G_A$ , representing states of the dynamic system (1) and (2), and digraph  $G_W$  defining the *agent communication network*. Let  $G_W = (V_W, E_W)$ , where  $V_W = \{1, \dots, N\}$  is the vertex set consisting of  $N$  agents,  $E_W = \{(i, j) \mid i \leftarrow j\}$  is the edge set, and  $\mathcal{D}_i = \{i\} \cup \{j \mid (i, j) \in E\}$  denote the extended neighborhood of agent  $i$ . Notice that, unlike many works in the literature we do not constrain  $G_W$  to be undirected. In fact, no assumption on the topology is considered here, as designing  $G_W$  is a contribution of this paper.

*Example:* To shed more light on this, we give an example here. Consider the flocking motion example given in the Fig. 1. The position, velocity or acceleration of every quad-copters can be considered as a state of the system. having ground robots as agents, the coordination law among them, typically following dynamics (1) [47], defines the dynamical system,  $A$ , and system digraph,  $G_A$ . The system outputs/measurements are the states tracked by the ground robots, and the communication network (to be designed) among these ground-robots is  $G_W$ .

### III. PROBLEM FORMULATION

We employ a variant of the Networked Kalman-type Estimator (NKE) proposed in [26], [27]. Let  $\hat{x}_{k|m}^i$  be the state estimate of agent  $i$  at time  $k$  given the outputs up to time  $m$ , ( $m \leq k$ ), from its neighboring agents,  $j \in \mathcal{D}_i$ . Each agent implements the following:

(i) *Predictor and state fusion:*

$$\hat{\mathbf{x}}_{k|k-1}^i = \sum_{j \in \mathcal{D}_i} w_{ij} A \hat{\mathbf{x}}_{k-1|k-1}^j \quad (8)$$

(ii) *Estimator and output fusion:*

$$\hat{\mathbf{x}}_{k|k}^i = \hat{\mathbf{x}}_{k|k-1}^i + K_k^i \sum_{j \in \mathcal{D}_i} C_j^T (y_k^j - C_j \hat{\mathbf{x}}_{k|k-1}^i) \quad (9)$$

where  $W = \{w_{ij}\}$  is the *state fusion* weight matrix such that  $w_{ij} \geq 0$  with  $\sum_{j \in \mathcal{D}_i} w_{ij} = 1$  ( $W$  is stochastic), and  $K_k^i$  is the local estimator gain at agent  $i$ .

*Remark 1:* Following are some useful remarks: (i) The diagonal entries of  $W$  are all nonzero, since every agent is in its own extended neighborhood and uses its own information. (ii) The first equation (8) is a local prediction fusion where each agent  $i$  fuses the neighboring estimates from time  $k-1$  and then implements a local predictor. (iii) In the second equation (9), each agent  $i$  updates its local prediction with an *innovation* term. We define this innovation as the difference between the state prediction of agent  $i$  and the state measurements obtained via agents,  $j \in \mathcal{D}_i$ . Adding this term, agent,  $i$ , makes its

final estimate,  $\hat{\mathbf{x}}_{k|k}^i$ , for the current time step. (iv) The protocol given in Eqs. (8)–(9) takes place at the same time-scale as the system dynamics, see Fig. 2. Notice that both Eqs. (8)–(9) can be combined into one equation; we give separate equations for the ease of explanation.

Let the estimation error at agent  $i$  and time  $k$  be defined as

$$\mathbf{e}_k^i = \mathbf{x}_{k|k} - \hat{\mathbf{x}}_{k|k}^i, \quad (10)$$

and let  $\mathbf{e}_k = [(\mathbf{e}_k^1)^T, \dots, (\mathbf{e}_k^N)^T]^T$  be the networked estimation error derived in the following.

*Proposition 1:* Let  $\mathbf{q}_k^i$  be some function of the system and measurement noise,  $\mathbf{v}_k$  and  $\mathbf{r}_k^i$ , independent of  $\mathbf{e}_{k-1}$  and let

$$\begin{aligned} K_k &= \text{blockdiag}[K_k^1, \dots, K_k^N], \\ D_C &= \begin{bmatrix} \sum_{j \in \mathcal{D}_1} C_j^T C_j & & 0 \\ & \ddots & \\ 0 & & \sum_{j \in \mathcal{D}_N} C_j^T C_j \end{bmatrix}, \\ \mathbf{q}_k &= [(\mathbf{q}_k^1)^T, \dots, (\mathbf{q}_k^N)^T]^T. \end{aligned} \quad (11)$$

Then we get the following networked error dynamics,

$$\mathbf{e}_k = (W \otimes A - K_k D_C (W \otimes A)) \mathbf{e}_{k-1} + \mathbf{q}_k. \quad (12)$$

The derivation requires some straightforward manipulations and is omitted here. Comparing this to Eq. (5), we note that the networked estimation error,  $\mathbf{e}_k$ , can be stabilized if and only if, the following pair,

$$(W \otimes A, D_C), \quad (13)$$

is observable. In other words, a gain matrix,  $K_k$ , exists such that  $\rho(W \otimes A - K_k D_C (W \otimes A)) < 1$  (i.e., it is a Schur matrix), if and only if  $(W \otimes A, D_C)$  is observable, where  $\rho(\cdot)$  denotes the spectral radius of a matrix. As it can be seen from Eq. (13), the communication network,  $W$ , plays a major role in distributed estimation as opposed to the multiple time-scale approach where  $W$  is irrelevant. The role of  $W$  in observability is because of the single time-scale nature of the estimator, see Fig. 2.

*Remark 2:*

- The variables  $D_C$  and  $K_k$  are block-diagonal matrices.
- Every block diagonal,  $\sum_{j \in \mathcal{D}_i} C_j^T C_j$ , in the matrix  $D_C$ , can be thought of as a representation of all of the measurements in the extended neighborhood of agent  $i$ .

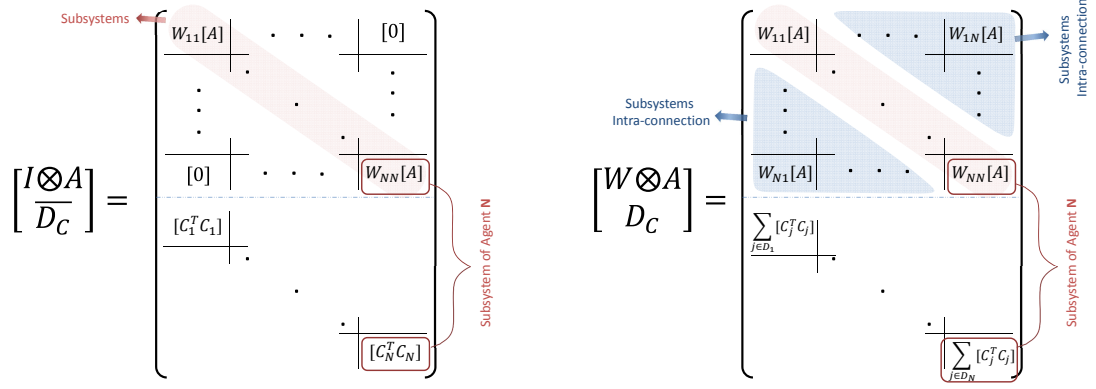


Fig. 4. (Left) Matrix structure of the distributed system with no data fusion. Every block diagonal  $W_{ii} \otimes A$  is a subsystem associated with the output/agent  $i$ . (Right) Adding data fusion, the intra-connections among these subsystems depends on the non-diagonals  $W_{ij} \otimes A$ ,  $i \neq j$ .

We refer to  $(W \otimes A, D_C)$  as the *distributed system* and  $G_{W \otimes A}$  as the graph associated with the matrix  $W \otimes A$ . For better understanding of the structural relevance of the estimator in (8)–(9), we first consider  $W = I$  and  $\overline{D}_C$  defined as follows,

$$\overline{D}_C = \begin{bmatrix} C_1^T C_1 & & \\ & \ddots & \\ & & C_N^T C_N \end{bmatrix}, \quad (14)$$

implying no information exchange among the agents. This distributed system,  $(I \otimes A, \overline{D}_C)$ , can be thought of as  $N$  subsystems each of them associated to an  $n \times n$  block diagonal in the matrix  $W \otimes A$ , see Fig.4 (Left). Now consider  $W$  to have some non-zero non-diagonal entries. As it is shown in Fig.4 (Right), these entries define the *inter-connections* among these subsystems.

To shed more light on this, consider the example given in Fig. 3 where we show a  $n = 7$ -state dynamical system with  $N = 3$  agents,  $\{a, b, c\}$  such that agent  $a$  measures  $x_3$ , agent  $b$  measures  $x_5$  and agent  $c$  measures  $x_7$ . Each agent is *required* to estimate the entire  $n = 7$  dimensional state-vector. Without any information fusion each agent only has a partial observation of the system as it is shown in Fig. 5. Therefore, each agent has to acquire the missing information via communicating with agents in its immediate neighborhood. However, in this illustration, no agent finds any observation in its neighborhood in addition to what it already possesses. Information sharing among the agents by applying state and output fusion provides more links among the subsystems in the distributed system digraph. This extra

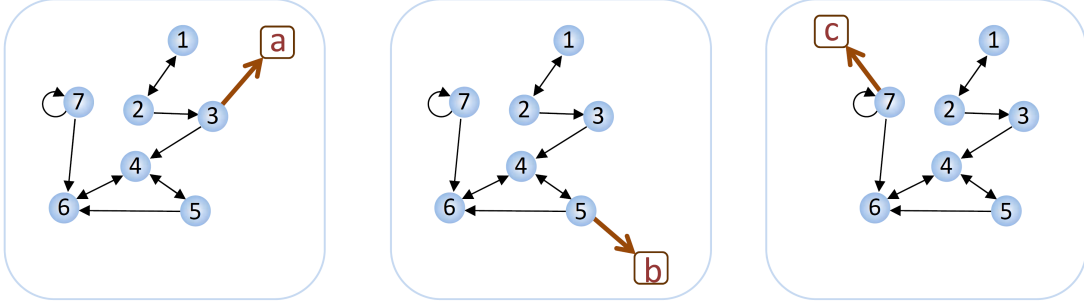


Fig. 5. The graph associated with distributed system,  $(I \otimes A, \overline{D_C})$ , with no data fusion represented as a sub-system at each agent. According to the Theorem 1, each sub-system (agent) is not observable with no data fusion.

linking among subsystems and outputs, captured by the non-zeros in  $W$  and the summation in  $D_C$ , has the *potential* to improve the generic observability of the system. In this regard, the main objective is to define the *structure* of the communication matrix  $W$  (graph  $G_W$ ) such that the distributed system  $(W \otimes A, D_C)$  is generically observable.

We seek a general method to make each subsystem observable. First, we describe how adding a link between two agents changes the graph structure of the distributed system. We explain this by considering the same example as in Fig. 3 and Fig. 5. In the case of output-fusion, a link between two agents, for example, from agent  $b$  to agent  $a$  ( $a \leftarrow b$ ), implies that agent  $a$  has access to agent  $b$ 's measurement, that is measurement of state  $x_5$ . However, the state-fusion case is more involved. For example, adding a link from agent  $b$  to agent  $a$  implies a nonzero entry in  $W$ , i.e.,  $w_{ab} \neq 0$ , which reflects in the networked system matrix,  $W \otimes A$ , as adding edges *to* some states in the subsystem associated to agent  $a$  *from* some states in the subsystem associated to agent  $b$ . This will be discussed in more detail in the next sections.

#### IV. OUR APPROACH

We now enlist our assumptions and provide a novel agent classification that will help in establishing our results.

##### A. Assumptions

In the rest of the paper, we make the following assumptions:

- (i) The communication between the agents is stable, i.e., the network is static;
- (ii) For every agent,  $i$ , the pairs,  $(A, C_i)$  or  $(A, \sum_{j \in \mathcal{D}_i} C_j^T C_j)$ , are not necessarily observable;

- (iii) The system is globally  $(A, C)$ -observable, i.e., if we collect all the sensor measurements at a center then the dynamical system is observable.

Assumption (ii), in practice, makes the networked estimation problem more challenging and is where this work becomes significantly different from current approaches, see, for example, [7] and references therein. Assumption (iii) is a typical assumption in distributed estimation implying the observability of centralized estimator; without this, no estimation scheme will work.

### B. State and Agent classification

To describe our approach, we provide a novel agent classification. Since the system is  $(A, C)$ -observable, (iii) in Section IV-A, we can enlist a disjoint set of cycles and  $Y$ -topped paths that covers all the state vertices (existence is ensured from condition (i)–Theorem 1). We are interested in a listing that involves the maximal cycles and we denote this set as  $\mathcal{L}$ . For example, from Fig. 3, the disjoint set of cycles and  $Y$ -topped paths that covers all the state vertices includes  $\{(4, 6, 4), (5, b), (1, 2, 3, a), (7, c)\}$ , and  $\{(4, 5, 6, 4), (1, 2, 1), (7, 7), (3, a)\}$ , among others. However, the latter includes the maximal cycles and thus  $\mathcal{L} = \{(4, 5, 6, 4), (1, 2, 1), (7, 7), (3, a)\}$ . The following classification is with respect to  $\mathcal{L}$ .

- (i) *Type- $\alpha$  agent* is an agent that appears in the  $Y$ -topped paths in  $\mathcal{L}$ . For example, agent  $a$  in Fig. 3.
- (ii) *Type- $\beta$  agent* is an agent that measures a state in the parent cycles in  $\mathcal{L}$ ; a parent cycle is a cycle that does not have an outgoing link to any other state not belonging to itself. For example, agent  $b$  in Fig. 3.
- (iii) *Type- $\gamma$  agent* is an agent that measure a state in the child cycles in  $\mathcal{L}$ ; a child cycle is a cycle that is not a parent cycle. For example, agent  $c$  in Fig. 3.

The above agent classification leads to the following definition and lemma.

*Definition 4 (Crucial observation):* A crucial observation is a measurement such that removing it renders the dynamical system unobservable.

*Lemma 2:* The agents of Type- $\alpha$  and Type- $\beta$  make “crucial” observations while the measurements at Type- $\gamma$  agents are not crucial.

*Proof:* Since the Type- $\beta$  agents monitor the parent cycles and there is no outgoing link from a parent cycle to any other state outside this cycle, the states in the parent cycles can only be the begin vertices of a  $Y$ -topped path (in order to satisfy condition (ii) in Theorem 1) when any one of these states is connected to an output. Hence all the Type- $\beta$  agents make crucial observations. On the other hand, removing a Type- $\alpha$  agent violates the condition (i) in Theorem 1 as the attached state vertex is not included in  $\mathcal{L}$  anymore. Hence, Type- $\alpha$  agents are also crucial. Finally, the only location for the Type- $\gamma$

TABLE I  
THIS TABLE SHOWS DISTRIBUTED SYSTEM FOR DIFFERENT FUSION LEVELS.

Fusion level	equivalent distributed system
No data fusion	$(I \otimes A, \overline{D_C})$
Only state fusion	$(W \otimes A, \overline{D_C})$
Only output fusion	$(I \otimes A, D_C)$
Both measurement and state fusion	$(W \otimes A, D_C)$

agent is monitoring a child SCC, which either has a directed path to a Type- $\alpha$  agent or to a Type- $\beta$  agent and hence is redundant. While the contribution of this state remains in  $\mathcal{L}$  due to the cycle present there. ■

For example, in Fig. 3, if either agent  $a$  or agent  $b$  is removed, then the system becomes unobservable. It can also be verified that agent  $c$  is non-crucial. Having defined types of agents, we note that the observability of the distributed system can be recovered via either  $W \otimes A$  matrix (state-fusion) or  $D_C$  (output-fusion). Here, we provide the *minimal* sufficient number of communication among the agents. Unlike our previous works [26], [27], we do not impose any constraint on the system matrix,  $A$ . Furthermore, the generic approach is further robust to uncertain systems and to *linearized* approximation of nonlinear models where the structure is fixed while the values are a function of the operating point [29].

## V. RECOVERING OBSERVABILITY

In this section, we first present some helpful results for the development of the paper and then find a general solution for  $(W \otimes A, D_C)$  observability. We first discuss the role of state fusion, related to the structure of matrix  $W$ , and then the role of output fusion, related to the structure of matrix  $D_C$  (see Table I). The reason is to get more intuitive and separate solutions for state and output fusion; obviously, in real applications if two agents are linked together they may share all of their information, including both their measurement and state estimates, to maximally improve their current state estimates. The results and proofs in this section are mainly graph theoretic that is a direct consequence of using the generic approach.

### A. Results on rank genericity

The result below follows from Theorem 1 and Lemma 1 as provided in Section II.

*Corollary 1 (Full  $S$ -rank):* A system matrix,  $A$ , is full  $S$ -rank if and only if its associated digraph has a disjoint union of cycles covering all the state vertices.

Notice that non-zero diagonals of a matrix can be represented as a disjoint union of self-cycles in its associated digraph. From Corollary 1 and by Remark 1 (Section III) we obtain the following result.

*Corollary 2:* The communication matrix,  $W$ , has a disjoint union of self-cycles and,

$$S\text{-rank}(W) = N. \quad (15)$$

This is always true because  $W$  has non-zero diagonals, i.e.,  $w_{ii} \neq 0, \forall i$ . Consequently, we state the following lemma for the networked system  $(W \otimes A)$ .

*Lemma 3:* For the communication matrix,  $W_{N \times N}$ , and system matrix,  $A_{n \times n}$ , the networked system  $W \otimes A$  is structurally full-rank if and only if  $A$  is structurally full-rank. Mathematically,

$$S\text{-rank}(W \otimes A) = N \times n \iff S\text{-rank}(A) = n \quad (16)$$

*Proof:* Recall that for two matrices,  $W$  and  $A$ ,

$$\text{rank}(W \otimes A) = \text{rank}(W) \times \text{rank}(A) \quad (17)$$

From Corollary 2, we have  $\text{rank}(W) = N$  for almost all numerical values, and for any full rank matrix  $A_{n \times n}$ , we have

$$\max(\text{rank}(W \otimes A)) = N \times n, \quad (18)$$

Based on the definition of the  $S$ -rank, we can conclude that  $(W \otimes A)$  is generically full rank for almost all choices of numerical values. This proves the necessity. On the other hand, if  $\text{rank}(A) < n$  for any choice of  $W$ , then we have

$$\max(\text{rank}(W \otimes A)) < N \times n, \quad (19)$$

which implies that,

$$S\text{-rank}(W \otimes A) < N \times n. \quad (20)$$

This proves the sufficiency. ■

## B. State fusion

We now explore Eq. (8) in NKE protocol and assume that there is no output fusion. In particular, we analyze the structure of the matrix  $W$  for  $(W \otimes A, \overline{D_C})$  observability according to Table I. First, we provide some special cases where the system matrix,  $A$ , is structurally full rank. This is the case, for

example, in linearization and discretization of non-linear systems where the system matrix *almost always* has non-zero diagonal entries.

*Lemma 4:* For full  $S$ -rank system matrix,  $(A, C)$  is centrally observable if and only if every parent SCC is output connected, i.e., monitored by (at least) one agent.

The proof is straightforward and omitted here. Interested readers may see our previous work in [27]. The following theorem establishes conditions on the communication network,  $G_W$ , over full  $S$ -rank systems.

*Theorem 2:* With a full  $S$ -rank system,  $A$ , the pair  $(W \otimes A, \overline{D_C})$  is generically observable when for every parent-SCC in  $A$ , say  $\mathcal{K}$ , if agent  $i$  does not have an observation of a state in  $\mathcal{K}$ , then in the communication network,  $G_W$ , there must be a directed path from agent  $i$  to any agent  $j$ , which has a state observation in<sup>2</sup>  $\mathcal{K}$ .

*Proof:* The system matrix  $A$  being full  $S$ -rank ensures the condition (i) in Theorem 1. This is because from Corollary 1, there exists a disjoint union of cycles alone that cover all of the state vertices and the  $Y$ -topped paths are not needed to verify condition (i). To satisfy condition (ii), all state vertices in a subsystem associated to every agent, say  $i$ , must be a begin vertex of a  $Y$ -topped path. This condition, according to Lemma 4, is satisfied by having every parent-SCC in  $W \otimes A$  be output-connected.

Since in communication matrix  $W$  there is a path from agent  $i$  to  $j$ , in  $G_{W \otimes A}$  graph, subsystem of agent  $i$  is connected to subsystem of agent  $j$ . Therefore, every state vertex in parent-SCC  $\mathcal{K}$  in subsystem  $i$  is connected to parent-SCC  $\mathcal{K}$  in subsystem  $j$  (see Fig. 6). Since every state of parent-SCC  $\mathcal{K}$  in subsystem  $j$  is  $Y$ -topped path to output  $j$ , every state of parent-SCC  $\mathcal{K}$  in subsystem  $i$  is also connected to output  $j$ . With this for every parent-SCC  $\mathcal{K}$  in every subsystem  $i$  of  $G_{W \otimes A}$ , all parent-SCCs are output connected and the theorem follows. ■

For example, consider again the three-output system in Fig. 5. Having vertices  $\{4, 5, 6\}$  as parent-SCC, agent  $b$  is the Type- $\alpha$  agent. According to the above theorem any other agent without any observation in  $\{4, 5, 6\}$ , like agent  $a$ , must be connected to agent  $b$ . This provides a connection from parent-SCC  $\{4, 5, 6\}$  in subsystem  $a$  to its counterpart SCC in subsystem  $b$  in distributed system graph  $G_{W \otimes A}$ , and in turn its output connectivity.

A very important point to mention here is that for full  $S$ -rank systems, there only exist Type- $\beta$  and Type- $\gamma$  agents. We prove this in the following lemma.

<sup>2</sup>If there is more than one agent observing SCC  $\mathcal{K}$ , say agents  $j, k$ , a directed path from agent  $i$  to only one of them is sufficient.



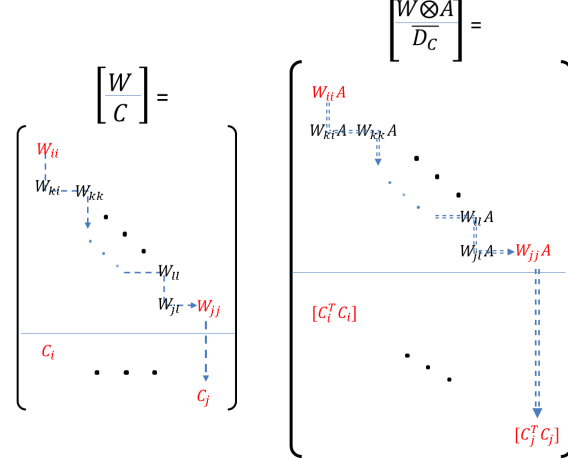


Fig. 6. This figure illustrates the proof of the Theorem 2 showing that: A directed path from agent  $i$  to agent  $j$  in  $W$  matrix (left) implies a directed path from states in subsystem  $i$  to subsystem  $j$  and consequently agent  $j$  in  $(W \otimes A)$  matrix.

*Lemma 5:* If a system matrix has full  $S$ -rank then we only have Type- $\beta$  or Type- $\gamma$  agents.

*Proof:* The proof is straightforward and relies on Corollary 1. Since  $A$  is full  $S$ -rank, there exists a set of disjoint cycles that covers all of the state vertices. Hence, the set  $\mathcal{L}$  introduced for agent classification in Section IV-B does not include any  $Y$ -topped paths and thus we cannot have any Type- $\alpha$  agent. ■ However, when the system matrix is not full  $S$ -rank, we also encounter Type- $\alpha$  agents that possess crucial observations from Lemma 2. It turns out that if the system matrix is not full  $S$ -rank, then even using a fully-connected communication network (complete  $G_W$  graph) does not recover observability. We now provide our main result on state fusion.

*Theorem 3:* Assume that  $(A, C_i)$  is not observable at any agent  $i$ . If system,  $A$ , is not full  $S$ -rank, then the NKE (8)–(9) is not observable with state fusion alone, i.e.,  $(W \otimes A, \overline{D_C})$  is not observable for any choice of the communication matrix  $W$ .

*Proof:* Let  $i$  be an agent for which condition (i) in Theorem 1 does not hold, i.e.,

$$S\text{-rank} \left( \begin{bmatrix} A \\ C_i^T C_i \end{bmatrix} \right) < n. \quad (21)$$

Such an agent always exists because: (i) based on the assumption (ii) in Section III, the entire system is not observable at any agent; and (ii) the matrix  $A$  is not full  $S$ -rank. Now consider  $(W \otimes A, \overline{D_C})$  for the best-case scenario where  $G_W$  is an all-to-all network. Let  $W_i$  be the  $i$ th column of the adjacency matrix  $W$ . Obviously,  $W_i \otimes A$  is the  $i$ th block column of  $(W \otimes A)$ , and contains block matrices  $W_{ji}A$

$$\begin{aligned}
& \begin{bmatrix} W_{1i}[A] \\ W_{2i}[A] \\ \vdots \\ W_{Ni}[A] \\ [C_i^T C_i] \end{bmatrix} \\
&= \begin{bmatrix} W_i \otimes A \\ C_i^T C_i \end{bmatrix} \\
& \quad \begin{bmatrix} W_{11}[A] & \dots & W_{1i}[A] & \dots & W_{1N}[A] \\ \vdots & \dots & \vdots & \dots & \vdots \\ W_{N1}[A] & \dots & W_{Ni}[A] & \dots & W_{NN}[A] \\ [C_1^T C_1] & \dots & [C_i^T C_i] & \dots & [C_N^T C_N] \end{bmatrix} \\
&= \begin{bmatrix} W \otimes A \\ \overline{D_C} \end{bmatrix}
\end{aligned}$$

Fig. 7. The figures illustrates the structure of  $W_i \otimes A$  (left) and matrix  $W \otimes A$  (right) in the proof of Theorem 3.

where  $W_{ji} \neq 0$  is the element in  $j$ th row and  $i$ th column of the full matrix  $W$ . It follows that

$$S\text{-rank} \left( \begin{bmatrix} W_{ji}A \\ C_i^T C_i \end{bmatrix} \right) < n, \quad (22)$$

for all  $j = 1, \dots, N$  as  $w_{ji} \neq 0$  and scalar multiplication does not change the structure and the  $S$ -rank (maximum possible rank over all values). Since  $A$  is not full  $S$ -rank,  $W_i \otimes A$  has rank less than  $n$  as stacking matrices with the same structure on top of each other (see Fig.7-Left) does not improve the  $S$ -rank, which immediately results in

$$S\text{-rank} \left( \begin{bmatrix} W_i \otimes A \\ C_i^T C_i \end{bmatrix} \right) < n. \quad (23)$$

Consequently, according to Fig.7-right, the structure of the matrix  $W \otimes A$  is given as the side-by-side concatenation of the matrices  $W_i \otimes A$ . Thus we have,

$$S\text{-rank} \left( \begin{bmatrix} W \otimes A \\ \overline{D_C} \end{bmatrix} \right) < Nn. \quad (24)$$

This holds for almost all choices of non-zero elements in the  $W$  matrices. Clearly, for any lower  $S$ -rank  $W$  the rank of  $W \otimes A$  cannot be recovered as well. Therefore, according to Lemma 1, the condition (i) in Theorem 1 is violated. ■

The above theorem shows that when  $A$  is  $S$ -rank deficient, then using state fusion cannot always guarantee the observability of the system and thus, the agents need access to more measurement data to

recover observability, which is discussed next. In contrast, Theorem 2 shows that when  $A$  is full  $S$ -rank, then state fusion alone can recover observability and further provides a method for the required agent communication. Clearly, these two results are to be viewed as a direct consequence of Assumptions (ii) and (iii) in Section IV-A. To the best of our knowledge, these conditions have not been developed before.

### C. Output fusion

The other solution to make the NKE observable is output fusion, that is the second update level given in the equation (9). According to the formulation, each agent shares its measurement with its direct neighbors and implements this as an innovation to update its prediction. According to Table I, for output fusion the structure of the matrix  $D_C$  has to be determined such that  $I \otimes A, D_C$  is observable.

Based on the definition of  $D_C$ , the  $i$ th  $n \times n$  diagonal block of  $D_C$  contains all of the measurements in the extended neighborhood of agent  $i$ . In the distributed system graph  $G_{(I \otimes A, D_C)}$ , say for the agent  $i$ , this is equivalent to adding all measurements in the neighborhood  $\mathcal{N}_i$  to the subsystem  $i$ .

We now provide our main result on output fusion.

*Theorem 4:* The system  $(I \otimes A, D_C)$  is observable if and only if:

- (i) The sub-graph of all Type- $\alpha$  and Type- $\beta$  agents is a complete graph, i.e. all these crucial agents are needed to be directly linked together;
- (ii) Every Type- $\gamma$  agent is directly connected to all Type- $\alpha$  and Type- $\beta$  agents.

*Proof:*

*Sufficiency:* With the given conditions (i) and (ii), each agent has access to all crucial measurements. This makes every agent generically observable.

*Necessity:* If an agent is not connected to one of the crucial agents, then it is missing a crucial measurement and the statement follows. ■

It can be verified that if a system is not  $(A, C)$  observable then even using a fully-connected communication network does not recover observability. Clearly, the only way to get a stable estimation error is by increasing the number of state observations and recovering the  $(A, C)$  observability. An interesting result on how to recover  $(A, C)$  observability can be found in [32].

### D. Main result

Finally, we consolidate our results in previous subsections on state and output fusion. Theorem 2 sets the condition for state fusion for full  $S$ -rank systems, i.e., conditions for  $(W \otimes A, \overline{D_C})$  generic observability, while Theorem 3 states that for general  $S$ -rank deficient systems networked observability

cannot be achieved via the state fusion alone. Output fusion, i.e., generic observability of  $(I \otimes A, D_C)$ , is discussed in Theorem 4. Combining these results, we now provide the main theorem on generic observability of the single time-scale NKE protocol in Eqs. (8)–(9).

*Theorem 5:* For  $(W \otimes A, D_C)$  observability with minimal number of communications, each agent needs:

- 1) A direct link *from* all the Type- $\alpha$  agents (output-fusion);
- 2) A directed path *to* (at least) one Type- $\beta$  agent for every parent SCC of  $A$ . This means, if there is two or more agents observing the same parent SCC, a directed path to any one of them is sufficient (state-fusion).

*Proof:* The proof is a direct consequence of Theorems 2, 3, and 4. ■

The following is a complementary remark to the Theorem 5.

*Remark 3:* In the case of Type- $\beta$  agents, every agent requires either a directed path *to* each Type- $\beta$  agent (as stated in the Theorem 5) or a direct link *from* each Type- $\beta$  agent (as stated in the Theorem 4); either one of these two conditions is sufficient for observability. However, the first strategy requires less number of links compared to the latter one, and therefore, it is preferred in terms of the minimal number of links in the communication network.

Notice that compared to the typical assumptions on the agents' network in the literature, like strong connectivity or having a cyclic path, here we provide milder condition on the non-crucial agents; as there is no need for connectivity *to* these agents but *from* these agents. Furthermore, an agent may have no measurement of the system and still be able to estimate the state of the system via the proposed strategies. Such agents, for example, may play a role to provide and maintain the connectivity of the agent communication network [48], or even, maintaining directed paths to Type- $\beta$  agents as stated in the second condition of the Theorem 5.

## VI. DESIGN OF LOCAL ESTIMATOR GAIN

In this subsection, we consider the design of the estimator gain matrix,  $K_k$ . Notice that having  $(W \otimes A, D_C)$  observable guarantees a *full* gain matrix,  $K_k$ , to stabilize the NKE error. However, according to protocol (9), we need a local gain matrix,  $K_k$ , which is *block-diagonal* with  $N$  blocks of  $n \times n$  matrices. For this section, we assume a constant estimator gain matrix is applied, i.e., the matrix  $K_k$  is independent of time,  $k$ , and denote it by  $K$ .

A partial list of references devoted to find constrained estimator gain for control and estimation is [26], [49]–[52]. Here, we use the Linear Matrix Inequality (LMI) approach in [50], [51]. However, in general,

the corresponding LMIs do not have a solution, because of the structural constraints (block-diagonal) on the gain matrix,  $K$ . This is the main difficulty in distributed estimation and control as convex/semidefinite approaches are not directly applicable. To this end, we implement an iterative procedure to solve LMIs under structural constraints.

In this regard, the following lemma presents the optimization approach to solve the estimator gain design problem. Interested readers may find more details in [26], [50].

*Lemma 6:* If the NKE protocol (8)–(9) is observable, then estimator gain matrix,  $K$ , is the solution of the following optimization problem.

$$\begin{aligned} \min \quad & \mathbf{trace}(XY) \\ \text{subject to} \quad & X, Y > 0, \\ & \begin{bmatrix} X & \hat{A}^T \\ \hat{A} & Y \end{bmatrix} > 0, \quad \begin{bmatrix} X & I \\ I & Y \end{bmatrix} > 0, \\ & K \text{ is block-diagonal.} \end{aligned} \tag{25}$$

where,

$$\hat{A} = W \otimes A - KD_C(W \otimes A) \tag{26}$$

In fact, we need a block-diagonal  $K$  such that  $\hat{A}$  is Schur (i.e.  $\rho(\hat{A}) < 1$ ).

Notice that, the solution to the second LMI is equivalent to  $X = Y^{-1}$ , which gives the minimum trace and the optimal value as  $nN$ . The nonlinear product of  $X$  and  $Y$  can be replaced with a linear approximation [50], [51], [53],  $\phi_{lin}(X, Y) = \mathbf{trace}(Y_0X + X_0S)$  and an iterative algorithm [51] can be used to minimize  $\mathbf{trace}(XY)$  under the given constraints.

---

**Algorithm 1** Iterative calculation of local gain estimator,  $K$ .

---

0: Find feasible points  $X_0, Y_0, K$ . If no such points exist, Terminate.

0: At iteration  $t > 0$  minimize  $\mathbf{trace}(Y_tX + X_tY)$  under the constraints given in (25) and find  $X, Y, K$ .

0: If  $\rho(\hat{A}) < 1$  terminate, otherwise set  $Y_{t+1} = Y$ ,  $X_{t+1} = X$  and run the step 2 for next iteration  $t = t + 1$ .

---

It is shown in [51] that  $\mathbf{trace}(Y_tX + X_tY)$  is a non-increasing sequence that converges to  $2nN$ . In this regard, a stopping criterion in step 3 of the above algorithm can also be established in terms of reaching within  $2nN + \epsilon$  of the trace objective. The iterative procedure given above is centralized, however, the center has to implement this process only once, off-line; then it transmits the estimator gains to each

agent and plays no further role in the implementation of local estimators; each agent, subsequently, observes and performs in-network operations to implement the estimator. A single time-scale algorithm can also be employed, where the above iterative procedure is implemented at the same time-scale  $k$  as of the dynamical system. With this approach, the estimator gain iterations,  $K_t$ , at each  $t$  is applied to the estimator at time-step,  $k$ , and may be transmitted to each agent at each step  $k$ . This is helpful when the implementation is assumed in real-time.

## VII. ILLUSTRATIVE EXAMPLE AND SIMULATION

Consider the system,  $(A, C)$ , given in Fig. 3. The structure of these matrices is given by

$$A = \begin{bmatrix} 0 & \times & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \times & 0 & \times & \times & 0 \\ 0 & 0 & 0 & \times & 0 & 0 & 0 \\ 0 & 0 & 0 & \times & \times & 0 & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & \times \end{bmatrix}, \quad (27)$$

$$\begin{bmatrix} C_a \\ C_b \\ C_c \end{bmatrix} = \begin{bmatrix} 0 & 0 & \times & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \times & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \times \end{bmatrix}. \quad (28)$$

Now, recall that based on Theorem 1, the system is globally observable by collection of the three measurements. The state partitioning with maximal cycles so that this partitioning covers all the states (in light of condition (i)–Theorem 1) is  $\mathcal{L} = \{(4, 5, 6, 4), (1, 2, 1), (7, 7), (3, a)\}$ . By definition, agent  $a$  is Type- $\alpha$ , agent  $b$  is Type- $\beta$  and agent  $c$  is Type- $\gamma$ . It can further be verified that agents  $a$  and  $b$  possess crucial observations.

To better illustrate the networking effect, we first note that the networked system with no information sharing, i.e., the graph associated to  $(I \otimes A, \overline{D_c})$ , is not observable at any of the agents individually (Fig. 5). To make the networked system observable at each agent, we propose the following communication matrices  $W_1$  and  $W_2$ , and their associated graphs  $G_W$  in Fig. 8 as two minimal networks making the system observable.

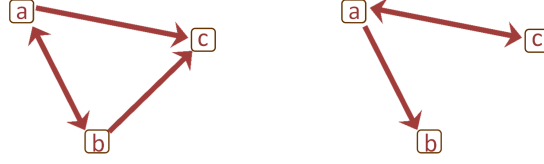


Fig. 8. (Left) communication network for the three agents using output fusion (Theorem 4), (Right) both measurement and state fusion (Theorem 5).

$$W_1 = \begin{bmatrix} \times & \times & 0 \\ \times & \times & 0 \\ \times & \times & \times \end{bmatrix}, \quad W_2 = \begin{bmatrix} \times & 0 & \times \\ \times & \times & 0 \\ \times & 0 & \times \end{bmatrix}. \quad (29)$$

The graph associated to  $W_1$  is proposed based on the Theorem 4. In this network crucial agents,  $\{a, b\}$ , are directly linked among each other and both have a directed link to agent,  $c$ , with no crucial observation. The second communication network  $W_2$  is based on Theorem 5; there is a direct link from agent  $a$  (Type- $\alpha$ ) to all other agents, and there is a path from every other agent to agent  $b$  (Type- $\beta$ ). It can be verified that for both topologies  $(W \otimes A, D_C)$  is generically observable. Note that the solution for the network design problem is not unique, and there maybe other examples of communication network satisfying the conditions in the last section. In addition, any network including one of these two topologies as a sub-graph is also a solution to the NKE problem. For example, under the full structured rank assumption of system  $A$ , any strongly connected network among agents suffices for individual observability [26].

For simulation, we consider a random valued matrix,  $A$ , with the structure in Eq. (27), and an output matrix, Eq. (28), with all non-zero entries equal to 1. The system eigenvalues are as follows.

$$\begin{aligned} \text{eig}(A) = [-1.0838, 1.0838, 0.6511, -0.5571, \dots \\ -0.0940, 0.0000, 1.3072] \end{aligned}$$

Clearly, the dynamical system is unstable, since  $\rho(A) = 1.0838$ . We choose the agents' network according to Fig. 8 (Right) with random link weights such that it remains stochastic. We use Algorithm 1 to find the block-diagonal gain matrix,  $K$ . Using this gain matrix, the eigen-values of the error dynamics, i.e.,

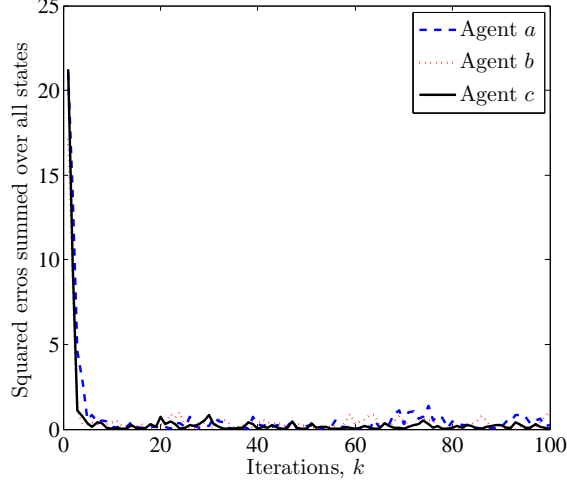


Fig. 9. Performance of the networked estimator at each agent. The error is squared and then summed over  $n = 7$  states at each iteration.

of the matrix,  $\hat{A}$ , are as follows,

$$\begin{aligned} \text{eig}(\hat{A}) = & [0.8190, 0.6511, -0.5571, 0.0073 \pm 0.3159i, \dots \\ & -0.2700, 0.1643, -0.1406, -0.0940, \\ & -0.0788, 0.0214, -0.0237, 0, 0, 0, 0, 0, 0, 0, 0], \end{aligned}$$

which are all stable. The system and output noise are, respectively,  $\mathbf{v}_k \sim \mathcal{N}(0, 0.05)$  and  $\mathbf{r}_k^i \sim \mathcal{N}(0, 0.2)$ . As system initial state we choose a random initial value between 0 and 3. The system error evolution over 100 iterations for three agents are given in Fig. 9. For the visual clarity, we have squared the errors at each iteration and then summed them over the  $n = 7$  states of the dynamical system. As it can be seen, the estimation error at all agents is bounded despite the fact that system is not stable.

## VIII. CONCLUSION

In this paper, we study the role of the agent communication network towards error stability of the NKE protocol (8)–(9) in the context of single time-scale distributed estimation. As opposed to multiple time-scale strategies where the communication network is irrelevant and diffusion strategies where the estimator error is *irrespective* of system dynamics, here, we take into account both system dynamics and communication network. We show that the NKE is able to track even potentially unstable dynamical systems, i.e., the networked estimator is observable for all stable and unstable eigenvalues. We show that under sufficient communication among the agents, the system state is generically observable at



every agent. Here, we provide minimal sufficient network connectivity applicable for multi-agent systems subject to constrained communication, e.g., out-of-range geographical distances or costly communication. We define three types of agents/measurement where two types are crucial for observability. We provide two main results on recovering networked observability: (i) with state-fusion, and (ii) with output-fusion. Furthermore, we determine dynamical systems ( $S$ -rank deficient) for which no state-fusion results in an observable networked estimator and one has to rely on output-fusion as well.

Our results are on the existence of a network structure for bounded estimation error and further finding such network with minimal links. Because of the *genericity*, the link weights are free parameters and results are independent of any particular fusion rule (e.g., Metropolis-Hastings [54]) chosen in (8)–(9). Nevertheless, the structure of the underlying agent communication remains relevant and leads to network/infrastructure design. Furthermore, link weights can be optimized to reduce the error, which is a direction for the future work. It is worth noting that, in general,  $S$ -rank and other generic properties are easily verified. For example, there are efficient graph theoretic, [28], flow theoretic, [55], and linear programming, [56], methods that can be employed to check for generic properties.

## REFERENCES

- [1] D. Acemoglu, A. Nedic, and A. Ozdaglar, “Convergence of rule-of-thumb learning rules in social networks,” in *47th IEEE Conference on Decision and Control*, Dec. 2008, pp. 1714–1720.
- [2] J. Cortes, “Distributed kriged kalman filter for spatial estimation,” *Automatic Control, IEEE Transactions on*, vol. 54, no. 12, pp. 2816–2827, dec. 2009.
- [3] H. Sayyaadi and M. Moarref, “A distributed algorithm for proportional task allocation in networks of mobile agents,” *Automatic Control, IEEE Transactions on*, vol. 56, no. 2, pp. 405–410, feb. 2011.
- [4] U. A. Khan and M. Doostmohammadian, “A sensor placement and network design paradigm for future smart grids,” in *4th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, San Juan, Puerto Rico, Dec. 2011, pp. 137–140.
- [5] Le Xie, Dae-Hyun Choi, and S. Kar, “Cooperative distributed state estimation: Local observability relaxed,” in *Power and Energy Society General Meeting, 2011 IEEE*, july 2011, pp. 1–11.
- [6] Jinghe Zhang, G. Welch, G. Bishop, and Zhenyu Huang, “Reduced measurement-space dynamic state estimation (remedyse) for power systems,” in *PowerTech, 2011 IEEE Trondheim*, june 2011, pp. 1–7.
- [7] R. Olfati-Saber and P. Jalalkamali, “Collaborative target tracking using distributed kalman filtering on mobile sensor networks,” in *30th IEEE American Control Conference*, San Francisco, CA, Jun. 2011.
- [8] H. Hashemipour, S. Roy, and A. Laub, “Decentralized structures for parallel Kalman filtering,” *IEEE Trans. on Automatic Control*, vol. 33, no. 1, pp. 88–94, Jan. 1988.
- [9] A. Mutambara, *Decentralized estimation and control for multisensor systems*, CRC Press, Boca Raton, FL, 1998.
- [10] F.S. Cattivelli and A.H. Sayed, “Diffusion strategies for distributed kalman filtering and smoothing,” *Automatic Control, IEEE Transactions on*, vol. 55, no. 9, pp. 2069–2084, sept. 2010.

- [11] F.S. Cattivelli and A.H. Sayed, “Distributed detection over adaptive networks using diffusion adaptation,” *Signal Processing, IEEE Transactions on*, vol. 59, no. 5, pp. 1917–1932, may 2011.
- [12] C.G. Lopes and A.H. Sayed, “Incremental adaptive strategies over distributed networks,” *Signal Processing, IEEE Transactions on*, vol. 55, no. 8, pp. 4064–4077, aug. 2007.
- [13] D. P Bertsekas, “Incremental gradient, subgradient, and proximal methods for convex optimization: A survey,” Tech. Rep., MIT, 2010, LIDS Technical Report.
- [14] M. Farina, G. Ferrari-Trecate, and R. Scattolini, “Distributed moving horizon estimation for linear constrained systems,” *Automatic Control, IEEE Transactions on*, vol. 55, no. 11, pp. 2462–2475, nov. 2010.
- [15] A. Ribeiro, G.B. Giannakis, and S.I. Roumeliotis, “Soi-kf: Distributed kalman filtering with low-cost communications using the sign of innovations,” *Signal Processing, IEEE Transactions on*, vol. 54, no. 12, pp. 4782–4795, dec. 2006.
- [16] G. Battistelli, L. Chisci, S. Morrocchi, and F. Papi, “An information-theoretic approach to distributed state estimation,” in *18th IFAC World Congress*, 2011, number 1, pp. 12477–12482.
- [17] Sheng-Yuan Tu and Ali H. Sayed, “Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks,” May. 2012, available at <http://arxiv.org/abs/0809.0009>.
- [18] J. Chen and A. H. Sayed, “Diffusion adaptation strategies for distributed optimization and learning over networks,” submitted for publication, available at <http://arxiv.org/abs/0809.0009>, Oct. 2011.
- [19] R. Olfati-Saber, “Distributed Kalman filters with embedded consensus filters,” in *44th IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005, pp. 8179–8184.
- [20] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, “Distributed Kalman filtering using consensus strategies,” in *Proceedings of the 46th IEEE Conference on Decision and Control*, 2007, pp. 5486–5491.
- [21] Usman A. Khan and José M. F. Moura, “Distributing the Kalman filter for large-scale systems,” *IEEE Transactions on Signal Processing*, vol. 56(1), no. 10, pp. 4919–4935, Oct. 2008.
- [22] E. J. Msechu, S. D. Roumeliotis, A. Ribeiro, and G. B. Giannakis, “Decentralized quantized Kalman filtering with scalable communication cost,” *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3727–3741.
- [23] U.A. Khan and A. Jadbabaie, “On the stability and optimality of distributed kalman filters with finite-time data fusion,” in *American Control Conference (ACC)*, 2011, 29 june- 1 july 2011, pp. 3405–3410.
- [24] U. A. Khan, S. Kar, A. Jadbabaie, and J. M. F. Moura, “On connectivity, observability, and stability in distributed estimation,” in *49th IEEE Conference on Decision and Control*, 2010, pp. 6639–6644.
- [25] S. Kar, J. M. F. Moura, and K. Ramanan, “Distributed parameter estimation in sensor networks: Nonlinear observation models and imperfect communication,” submitted for publication, available at <http://arxiv.org/abs/0809.0009>, Aug. 2008.
- [26] U. A. Khan and A. Jadbabaie, “Coordinated networked estimation strategies using structured systems theory,” in *49th IEEE Conference on Decision and Control*, Orlando, FL, Dec. 2011, pp. 2112–2117.
- [27] M. Doostmohammadian and U. A. Khan, “Communication strategies to ensure generic networked observability in multi-agent systems,” in *45th Annual Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2011, pp. 1865–1868.
- [28] K. J. Reinschke, *Multivariable control, a graph theoretic approach*, Berlin: Springer, 1988.
- [29] Y.Y. Liu, J.J. Slotine, and A.L. Barabasi, “Controllability of complex networks,” *Nature*, vol. 473, no. 7346, pp. 167–173, May 2011.
- [30] T. Boukhboza, F. Hamelin, S. Martinez-Martinez, and D. Sauter, “Structural analysis of the partial state and input

- observability for structured linear systems: Application to distributed systems,” *European Journal of Control*, vol. 15, no. 5, pp. 503–516, Oct. 2009.
- [31] J.-M. Dion, C. Commault, and J. van der Woude, “Generic properties and control of linear structured systems: a survey,” *Automatica*, vol. 39, pp. 1125–1144, Mar. 2003.
  - [32] T. Boukhobza and F. Hamelin, “State and input observability recovering by additional sensor implementation: A graph-theoretic approach,” *Automatica*, vol. 45, no. 7, pp. 1737 – 1742, 2009.
  - [33] M. Ji and M. Egerstedt, “Observability and estimation in distributed sensor networks,” in *Decision and Control, 2007 46th IEEE Conference on*, dec. 2007, pp. 4221 –4226.
  - [34] M. Egerstedt, “Complex networks: Degrees of control,” *Nature*, vol. 473, pp. 158–159, May 2011.
  - [35] S. Sundhar Ram, A. Nedi, and V.V. Veeravalli, “Distributed stochastic subgradient projection algorithms for convex optimization,” *Journal of Optimization Theory and Applications*, vol. 147, pp. 516–545, 2010.
  - [36] P. Bianchi, G. Fort, W. Hachem, and J. Jakubowicz, “Convergence of a distributed parameter estimator for sensor networks with local averaging of the estimates,” in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, may 2011, pp. 3764 –3767.
  - [37] I.D. Schizas, G. Mateos, and G.B. Giannakis, “Distributed lms for consensus-based in-network adaptive processing,” *Signal Processing, IEEE Transactions on*, vol. 57, no. 6, pp. 2365 –2382, june 2009.
  - [38] W. Abbas and M. Egerstedt, “Hierarchical assembly of leader-asymmetric, single-leader networks,” in *American Control Conference (ACC) 2011*, San Francisco, CA, June 29-July 01 2011.
  - [39] R. Kalman and R.Bucy, “New results in linear filtering and prediction theory,” *ASME Journal of Basic Engineering*, vol. 83, pp. 95–108, 1961.
  - [40] M.L.J. Hautus, “Controllability and observability conditions of linear autonomous systems,” *Ned. Akad. Wetenschappen*, vol. Ser. A, no. 72, pp. 443–448, 1969.
  - [41] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, 209.
  - [42] E. J. Davison and S. H. Wang, “Properties of linear time-invariant multivariable systems subject to arbitrary output and state feedback,” *IEEE Transactions on Automatic Control*, vol. 18, no. 1, pp. 24–32, Feb. 1973.
  - [43] E. J. Davison and S. H. Wang, “Properties and calculation of transmission zeros of linear multivariable systems,” *Automatica*, vol. 10, pp. 643–658, 1974.
  - [44] F. Harary, “The determinant of the adjacency matrix of a graph,” *SIAM Review*, vol. 4, no. 3, pp. 202 –210, Jul. 1962.
  - [45] C. Lin, “Structural controllability,” *Automatic Control, IEEE Transactions on*, vol. 19, no. 3, pp. 201 – 208, jun 1974.
  - [46] S. Sundaram and C.N. Hadjicostis, “Distributed function calculation and consensus using linear iterative strategies,” *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 4, pp. 650 –660, may 2008.
  - [47] V.D. Blondel, J.M. Hendrickx, A. Olshevsky, and J.N. Tsitsiklis, “Convergence in multiagent coordination, consensus, and flocking,” in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC ’05. 44th IEEE Conference on*, dec. 2005, pp. 2996 – 3000.
  - [48] A. N. Kopeikin, S. S. Ponda, L. B. Johnson, O. Toupet, and J. P. How, “Real-time dynamic planning to maintain network connectivity in a team of heterogeneous unmanned systems,” in *Wi-UAV 2011, 2nd International Workshop on Wireless Networking for Unmanned Autonomous Vehicles at the IEEE GlobeComm Conference*, Dec 2011.
  - [49] Khan U. A. and A. Jadbabaie, “Networked estimation under information constraints,” submitted, available at <http://arxiv.org/abs/1111.4580>.

- [50] M. Pajic, S. Sundaram, J. Le Ny, G.J. Pappas, and R. Mangharam, “The wireless control network: Synthesis and robustness,” in *Decision and Control (CDC), 2010 49th IEEE Conference on*, 2010, pp. 7576–7581.
- [51] L. El Ghaoui, F. Oustry, and M. Ait Rami, “A cone complementarity linearization algorithm for static output-feedback and related problems,” *IEEE Trans. on Automatic Control*, vol. 42, no. 8, pp. 1171–1176, Aug. 1997.
- [52] A.I. Zecevic and D.D. Siljak, “Control design with arbitrary information structure constraints,” *Automatica*, vol. 44, no. 10, pp. 2642 – 2647, 2008.
- [53] O. L. Mangasarian and Jong-Shi Pang, “The extended linear complementarity problem,” *SIAM Journal on Matrix Analysis and Applications*, vol. 2, pp. 359–368, Jan. 1995.
- [54] Lin Xiao, Stephen Boyd, and Seung-yeon Kim, “Distributed average consensus with least-mean-square deviation,” *Journal of Parallel and Distributed Computing*, vol. 67, pp. 33–46, 2005.
- [55] V. Hovelaque, C. Commault, and J.-M. Dion, “Analysis of linear structured systems using a primal-dual algorithm,” *Syst. Control Lett.*, vol. 27, pp. 73–85, February 1996.
- [56] S. Poljak, “Maximum rank of powers of a matrix of a given pattern,” *Proceedings of the American Mathematical Society*, vol. 106, no. 4, pp. 1137–1144, 1989.